Linux BSP for the Freescale Lite5200B EVB

User's Guide

Rev. 2.0

07/2006

freescale
semiconductor

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document. Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Learn More: For more information about Freescale products, please visit www.freescale.com.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.
© Freescale Semiconductor, Inc. 2006. All rights reserved.

Contents

About This Manual 1-1

Audience 1-1

Organization 1-1

Conventions 1-1

Definitions, Acronyms, and Abbreviations 1-1

Chapter 1 Introduction 1-2

1.1 LTIB Overview 1-2

1.2 BSP Overview 1-2

Chapter 2 LTIB Basics 2-1

2.1 Installing the BSP 2-1

2.2 Running LTIB 2-1

Chapter 3 Target Configuration 3-1

3.1 Supported Target Revisions 3-1

3.2 Target Set-up 3-1

Chapter 4 Target Deployment 4-1

4.1 Host Set-up 4-1

4.2 Configuring u-boot 4-2

4.3 Development Depoyment 4-3

4.4 Production Deployment 4-3

4.5 Flashing u-boot 4-3

About This Manual

This User Manual provides information on the basic features supported by the BSP and provides you with instructions about how to accomplish these tasks:

- Install the BSP on a host development system.
- Run Linux Target Image Builder (LTIB) to build target images.
- Deploy built images to the Lite5200B EVB.
- Boot Linux on the Lite5200B EVB.

Audience

This document is addressed to developers who want to take advantage of the Freescale Linux Target Image Builder (LTIB) for the Lite5200B EVB Board Support Package (BSP).

Organization

This document is organized into 4 chapters.

Chapter 1	Provides an introduction to the Lite5200B EVB BSP.
Chapter 2	Provides basic information on LTIB
Chapter 3	Provides important target set-up information
Chapter 4	Provides host and target-specific build and deployment information

Conventions

This document uses the following notational conventions:

- Courier monospaced type indicates commands, command parameters, code examples, expressions, data types, and directives.
- Italic type indicates replaceable command parameters.
- All source code examples are in C.

Definitions, Acronyms, and Abbreviations

The following list defines the abbreviations used in this document.

BSP	Board Support Package
LTIB	Linux Target Image Builder
NFS	Network File System
TFTP	Trivial File Transfer Protocol

Chapter 1
Introduction

1.1 LTIB Overview

The Linux Target Image Builder (LTIB) is a tools framework used to manage, configure, extend and build Linux software elements to easily build a Linux target image and a root filesystem. LTIB runs on an x86 PC running the Linux OS.

This BSP operates with LTIB running on a host development system with the following:

- Ethernet card
- Serial port
- 1 GB of free disk space required
- NFS Server
- TFTP Server
- rsync
- perl

NOTE: Be aware that some host side packages may not function properly on every Linux distribution. The following are platforms where LTIB was tested.

- Redhat: 7.3, 8.0, 9.0
- Fedora Core: 1, 2, 3, 4
- Debian: 3.1r0 (stable), unstable
- SuSE: 8.2, 9.2, 10.0

1.2 BSP Overview

This Lite5200B EVB BSP is designed for use with LTIB. Once the BSP is installed and running with its basic configuration, you can use LTIB to customize your project.

The BSP components provide the tools, device drivers, and additional features needed for your embedded Linux project.

Pre-built Linux-2.6.16.11 kernel and packages with these features:

- Ingo Molnar's realtime preemption patch
- On-chip Serial port
- FEC
- Memory devices, including flash and DRAM
- Controller Area Network
- I2C Controller
- USB Host (Tested with Mouse and Keyboard)
- ATA/IDE Controller (PIO and UDMA)
- PCI Bus (Tested with CoralPA card and Realtek 8139)

U-boot 1.1.2
Toolchain (gcc 3.4.3, glib 2.3.6, binutils 2.16.1)

GDB 6.3
Documentation. See START_HERE.htm on this CD.

Chapter 2
LTIB Basics

2.1 Installing the BSP

Please follow the steps below to install LTIB on your host machine.

1. As root, mount the ISO image on your machine:
   `mount –o loop <target-bsp.iso> <mount point>`
2. As a non-root user, install the LTIB:
   `<mount point>/install`

The BSP EULA will be presented for acceptance. To continue install accept the licesnse. Then, input the desired LTIB install path. Be sure the user has the correct permissions for the install path..

There are no uninstall scripts. To uninstall LTIB you need to remove the `/opt/freescale/pkgs`, `/opt/freescale/ltib` and `/ltib` directories manually.

2.2 Running LTIB

To run LTIB, change to the directory into which you installed it and run `./ltib`.

```
cd /<ltib install dir>/

./ltib
```

The first time LTIB runs on your machine a number of host packages are built and installed that support LTIB. This may take a few minutes.

Important Note: Please be sure to set the "Target System Configuration" options for your network environment the first time you build.

To modify the project configuration simply run:

`./ltib --configure` (or `-c`; type `--help` to see configuration options)

This will re-prompt you for the platform/board configuration. In the board configuration screens, change settings and select packages as appropriate. When you exit the configuration screen your target image will be adjusted accordingly.

Once you build your project you will get following directory/image files:

- **rootfs/** – directory, the root file system that will be deployed on your board.
- **rootfs/boot/uImage** – kernel image that can be loaded with U-boot

If you want to fully re-configure and re-compile all the packages, you can do the following. This is generally not necessary.

1. Clean up all the configure files and objects thoroughly:

   ```
   ./ltib –m distclean
   ```

2. You will be prompted to confirm your choice. Type yes to perform a distclean.

3. Run litb

   ```
   ./ltib
   ```

More information on LITB can be found in *<install path>/<ltib install dir>*/doc. Or on the web at http://savannah.nongnu.org/projects/ltib.

Chapter 3
Target Configuration

3.1 Supported Target Revisions

The target system is the Lite5200B EVB board. This BSP is known to work on the following board(s) revision(s):

Lite5200B EVB: Rev K
CoralP Card: Rev 5.0

3.2 Target Set-up

1. Disconnect the power supply from the target and set the board configuration switches.

CFG 0	CFG 0	CFG 0	CFG 0	CFG 0
L	H	H	H	L

XLB	SYS	FVCO	MG	LF
L	L	L	L	L

HI/LO	WAIT	SWAO	WUDE	MUXED
H	H	L	L	L

2. Connect your board to the network via the Ethernet port.

3. Connect your board to your host machine via the serial port. For Linux, use the serial port marked "UART" next to the Ethernet jack.

4. Verify the dip switches are set correctly.

Chapter 4
Target Deployment

4.1 Host Set-up

Host setup is critical for your BSP to function. The host must be running tftp and nfs in order for deployment to work. The following instructions are generic. Your system may be different and the commands should be adjusted accordingly.

1. Turn off firewall for tftp to work. `iptables -F` or type "`setup`" at the command line.
2. Install tftp-server
3. Install nfs-server
4. Create the tftboot directory.

   ```
   mkdir /tftpboot
   ```

5. Link `rootfs` to an exportable directory once you have built your project.

   ```
   ln –s /<ltib install dir>/rootfs /tftpboot/ltib
   ```

6. Copy over kernel, bootloader, and flash filesystem images for your deployment to the `/tftpboot` directory

   ```
   cp /<ltib install dir>/rootfs/boot/uImage /tftpboot
   cp /<ltib install dir>/rootfs.jffs2 /tftpboot
   cp /<ltib install dir>/rootfs/boot/u-boot.bin /tftpboot
   ```

7. Edit `/etc/exports` and add the following line:

   ```
   /tftpboot/<ltib install dir>/ <target board IP>(rw,no_root_squash,
   async)
   ```

8. Edit `/etc/xinetd.d/tftp` to enable tftp like this:

   ```
   {
   disable     = no
   socket_type = dgram
   protocol    = udp
   wait        = yes
   user        = root
   server      = /usr/sbin/in.tftpd
   server_args = /tftpboot
    }
   ```

9. Restart the nfs and tftp servers on your host

   ```
   /etc/init.d/xinetd restart
   /etc/init.d/nfsserver restart
   ```

10. Connect board to the network
11. Connect the target to the host via a serial connection
12. Start `minicom` and set it up to talk to the target
    - Serial Setup: Select correct serial device; Hardware & Software Flow control = No; Bps = 115,200
    - Modem & dialing: Delete text for the following: Init String, Reset String, Hang-up String, No flow control
13. Power on board and see the console prompt.

4.2 Configuring u-boot

The factory installed u-boot has several commands predefined in the default environment. To verify that the u-boot you are using has these type 'print' at the u-boot prompt and you should see something like:

```
bootcmd=run flash_self
bootdelay=5
baudrate=115200
preboot=echo Type "run tftp_nfs" load kernel via tftp with nfs root;echo
kernel_flash_addr=ffd00000
kernel_ram_addr=200000
kernel_flash_size=200000
jffs2args=setenv bootargs console=ttyPSC0,115200 root=/dev/mtdblock0 rw
rootfstype=jffs2
nfsargs=setenv bootargs console=ttyPSC0,115200 root=/dev/nfs rw
nfsroot=$(serverip):$(rootpath)
netdev=eth0
addip=setenv bootargs $(bootargs)
ip=$(ipaddr):$(serverip):$(gatewayip):$(netmask):$(hostname):$(netdev):off
panic=1
tftp_nfs=tftp $(kernel_ram_addr) $(bootfile);run nfsargs addip;bootm
cpkernel=cp $(kernel_flash_addr) $(kernel_ram_addr) $(kernel_flash_size)
flash_nfs=run nfsargs addip cpkernel; bootm $(kernel_ram_addr))
flash_jffs2=run jffs2args cpkernel; bootm $(kernel_ram_addr)
bootcmd=run tftp_nfs
```

If the settings are wrong or missing type them in like this:

```
setenv preboot 'echo Type "run tftp_nfs" to tfp kernel with nfs root;echo'
setenv kernel_flash_addr ffd00000
setenv kernel_ram_addr 200000
setenv kernel_flash_size 200000
setenv jffs2args 'setenv bootargs console=ttyPSC0,115200 \
      root=/dev/mtdblock0 rw rootfstype=jffs2'
setenv nfsargs  'setenv bootargs console=ttyPSC0,115200 \
      root=/dev/nfs rw nfsroot=$(serverip):$(rootpath)'
setenv netdev eth0
setenv addip 'setenv bootargs $(bootargs) \
      ip=$(ipaddr):$(serverip):$(gatewayip):$(netmask):$(hostname):\
      $(netdev):off panic=1'
setenv tftp_nfs 'tftp $(kernel_ram_addr) $(bootfile);run nfsargs addip;bootm'
setenv cpkernel 'cp $(kernel_flash_addr) $(kernel_ram_addr) \
      $(kernel_flash_size)'
setenv flash_nfs 'run nfsargs addip cpkernel; bootm $(kernel_ram_addr))'
```

```
setenv flash_jffs2 'run jffs2args cpkernel; bootm $(kernel_ram_addr)'
setenv bootcmd 'run tftp_nfs'
```

You must set the console to ttyPSC0,115200 for the Lite5200B EVB

After verifying the above commands are in the u-boot environment proceed with the following steps to configure u-boot.

1. Set the board's network configuration

   ```
   setenv ipaddr 172.27.152.99
   setenv serverip 172.27.152.5
   setenv gatewayip 172.27.255.254
   setenv netmask 255.255.0.0
   ```

2. Set some pathnames needed later

   ```
   setenv rootpath /tftpboot/ltib
   setenv bootfile /tftpboot/uImage
   ```

3. Save the configuration to flash

   ```
   saveenv
   ```

4.3 Development Depoyment

1. Copy the kernel image to the tftpboot directory

   (On the host) `cp /<ltib install dir>/rootfs/boot/uImage /tftpboot`

2. Link `rootfs` to an exportable directory

   `ln –s /<ltib install dir>/rootfs /tftpboot/ltib`

3. Boot the kernel

   ```
   run tftp_nfs
   ```

   To automate this set the bootcmd variable:

   ```
   setenv bootcmd run tftp_nfs
   saveenv
   ```

4.4 Production Deployment

1. Copy the kernel image to the tftpboot directory

   (On the host) `cp /<ltib install dir>/rootfs/boot/uImage /tftpboot`

2. Download kernel and write it to flash

   ```
   tftp 1000000 /tftpboot/uImage
   erase ffd00000 ffefffff
   cp.b 1000000 ffd00000 $(filesize)
   ```

3. Split the jffs2 flash image into two files if it is larger than 16M.

(On the host) `split -a 1 -d -b 16m rootfs.jffs2 rootfs.jffs2_`

This will create a file called rootfs.jffs2_0. A second file called rootfs.jffs2_1 will also be created if the rootfs image is greater than 16M in size.

4. Copy the split jffs2 flash image files to the tftpboot directory

   (On the host) `cp /<ltib install dir>/rootfs.jffs2_* /tftpboot`

5. Erase the filesystem flash

```
erase fe000000 fefffffff
erase ff000000 ffcfffff
```

6. Download and flash the first 16M of the jffs2 flash image (this can take almost 10 minutes)

```
tftp 1000000 /tftpboot/rootfs.jffs2_0
cp.b 1000000 fe000000 $(filesize)
```

7. Download and flash the remainder of the jffs2 flash image (only if the image is larger than 16M)

```
tftp 1000000 /tftpboot/rootfs.jffs2_1
cp.b 1000000 ff000000 $(filesize)
```

8. Boot kernel in flash with jffs2 filesystem. The first time you boot jffs2 it will take some time.

```
run flash_jffs2
```

   To have this automatically happen at boot set bootcmd

```
set bootcmd run flash_jffs2
saveenv
```

4.5 Flashing u-boot

To restore the factory u-boot move jumper J10 (BOOT SEL) to the B/U position and reset the board. When the board boots the factory default u-boot image will be copied to the main flash. When complete you will be instructed to move the jumper back to the MAIN position.

To replace u-boot with one compiled with ltib follow these steps:

1. Setup u-boot networking parameters (See Section 4.3)
2. Copy the u-boot binary image to the tftpboot directory

   (On the host) `cp /<ltib install dir>/rootfs/boot/u-boot.bin /tftpboot`

3. Download u-boot via tftp

```
tftp 1000000 /tftpboot/u-boot.bin
```

4. Save u-boot to flash

```
protect off fff00000 fff5ffff
erase fff00000 fff5ffff
cp.b 1000000 fff00000 $(filesize)
```